Challenges of Symbolic Computation
My Favorite Open Problems

Erich Kaltofen
North Carolina State University
www.math.ncsu.edu/~kaltofen

Brief History

The ages of symbolic computation

60s: pioneering years: polynomial arithmetic, integration

70s: Macsyma; abstract domains: Scratchpad/II, Axiom

80s: polynomial-time methods; user interfaces: Mathematica

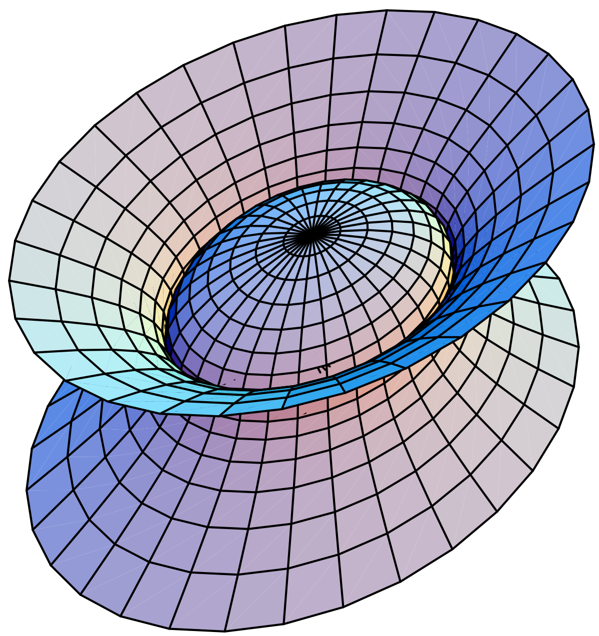
90s: teaching of calculus (Maple), math on the web

00s: merging of symbolic, numeric, and geometric paradigm (?)

1. Numeric/Symbolic

Factorization of nearby polynomials over the complex numbers

$$81x^4 + 16y^4 - 648z^4 + 72x^2y^2 - 648x^2 - 288y^2 + 1296 = 0$$



$$(9x^2 + 4y^2 + 18\sqrt{2}z^2 - 36)(9x^2 + 4y^2 - 18\sqrt{2}z^2 - 36) = 0$$

$$\begin{aligned} 81x^4 + 16y^4 - 648.003z^4 + 72x^2y^2 + .002x^2z^2 + .001y^2z^2 \\ - 648x^2 - 288y^2 - .007z^2 + 1296 = 0 \end{aligned}$$

Open Problem 1

Given is a polynomial $f(x,y) \in \mathbb{Q}[x,y]$ *and* $\varepsilon \in \mathbb{Q}$.

Decide in polynomial time in the degree and coefficient size if there is a factorizable $\hat{f}(x,y) \in \mathbb{C}[x,y]$ *with* $\|f - \hat{f}\| \leq \varepsilon$,

for a reasonable coefficient vector norm $\|\cdot\|$.

Sensitivity analysis: approximate consistent linear system

Suppose the linear system $Ax = b$ is unsolvable.
Find $\hat{b}$ "nearest to" b that makes it solvable.

Minimizing Euclidean distance: $\min_{\hat{x}} \|A\hat{x} - b\|_2$ (least squares)

Minimizing component-wise distance: $\min_{\hat{x}} \left(\max_{1 \le i \le m} \left| b_i - \sum_{j=1}^{n} a_{i,j}\hat{x}_j \right| \right)$

Introduce new variable y and solve the linear program

minimize: y

linear constraints: $y \ge b_i - \sum_{j=1}^{n} a_{i,j}\hat{x}_j \quad (1 \le i \le m)$

$y \ge -b_i + \sum_{j=1}^{n} a_{i,j}\hat{x}_j \quad (1 \le i \le m)$

Sensitivity analysis: nearest singular matrix

Given are $2n^2$ rational numbers $\underline{a}_{i,j}, \bar{a}_{i,j}$.
Let $\mathcal{A}$ be the *interval* matrix

$$\mathcal{A} = \left\{ \begin{bmatrix} a_{1,1} & \dots & a_{n,n} \\ \vdots & & \vdots \\ a_{n,1} & \dots & a_{n,n} \end{bmatrix} \mid \underline{a}_{i,j} \leq a_{i,j} \leq \bar{a}_{i,j} \text{ for all } 1 \leq i, j \leq n \right\}.$$

Does $\mathcal{A}$ contain a singular matrix?
This problem is *NP-complete* [Poljak&Rohn 1990].

When the distance is measured by a *matrix norm*, the problem can be solved efficiently [Eckart&Young 1936].

Sensitivity analysis: approximate greatest common divisor

Suppose $f = x^m + a_{m-1}x^{m-1} + \cdots + a_0$, $g = x^n + b_{n-1}x^{n-1} + \cdots + b_0$ have no common divisor.
Find $\hat{f}, \hat{g}$ "nearest to" f, g that have a common root.

Karmarkar&Lakshman [1996] minimize

$$\sqrt{|a_m - \hat{a}_m|^2 + \cdots + |a_0 - \hat{a}_0|^2 + |b_n - \hat{b}_n|^2 + \cdots + |b_0 - \hat{b}_0|^2}.$$

Equivalent formulation:
Compute the nearest singular *Sylvester matrix* to the Sylvester matrix

$$\begin{bmatrix} a_m & a_{m-1} & \cdots\cdots & a_0 & & & \\ & a_m & \cdots\cdots & a_1 & a_0 & & \\ & & \ddots & & \ddots & \ddots & \\ & & & a_m & \cdots\cdots\cdots & & a_0 \\ b_n & b_{n-1} & \cdots\cdots & b_0 & & & \\ & b_n & \cdots\cdots & b_1 & b_0 & & \\ & & \ddots & & \ddots & \ddots & \\ & & & b_n & \cdots\cdots\cdots & & b_0 \end{bmatrix}$$

Sensitivity analysis: Kharitonov [1978] theorem

Given are $2n$ rational numbers $\underline{a}_i, \bar{a}_i$.
Let P be the *interval* polynomial

$$P = \{x^n + a_{n-1}x^{n-1} + \cdots + a_0 \mid \underline{a}_i \le a_i \le \bar{a}_i \text{ for all } 0 \le i < n\}.$$

Then every polynomial in P is *Hurwitz* (all roots have negative real parts), if and only if the four “corner” polynomials

$$g_k(x) + h_l(x) \in P, \quad \text{where } k = 1,2 \text{ and } l = 1,2,$$

with

$$g_1(x) = \underline{a}_0 + \bar{a}_2x^2 + \underline{a}_4x^4 + \cdots, \quad h_1(x) = \underline{a}_1 + \bar{a}_3x^3 + \underline{a}_5x^5 + \cdots,$$
$$g_2(x) = \bar{a}_0 + \underline{a}_2x^2 + \bar{a}_4x^4 + \cdots, \quad h_2(x) = \bar{a}_1 + \underline{a}_3x^3 + \bar{a}_5x^5 + \cdots$$

are Hurwitz.

Sensitivity analysis: constraint root problem

Given is a real or complex polynomial

$$f(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0$$

and a root $\alpha \in \mathbb{C}$.

Compute $\hat{f}$ "nearest to" f such that $\hat{f}(\alpha) = 0$.

Hitz and K [1998] solve this problem efficiently for

- parametric α (root stability) and Euclidean distance
- explicit roots $\alpha_1, \alpha_2, \ldots$ and coefficient-wise distance
- with linear coefficient constraints, e.g., $a_n = 1$.

Symbolic and numeric computation: a marriage made in heaven?

2. Quantifier elimination (QE)

A simple QE problem over the real numbers

$$\text{for } a > 0\text{: } \min_x(ax^2+bx+c) \Leftrightarrow \forall y\colon a > 0 \text{ and } ay^2+by+c \geq ax^2+bx+c$$

$$\Leftrightarrow a > 0 \text{ and } x = -\frac{b}{2a}$$

The quantified variables can be eliminated; the values of the un-quantified variables that satisfy the expression form a *semi-algebraic* set.

QE is computable [Tarski 1948; Collins 1976; Grigoriev 1986; Hong 1990]

Open Problem 2 (Solotareff's problem by Collins 1992)
Eliminate the quantifiers and solve for $n \geq 6$ on a computer:

$$\text{for } r > 0\text{:} \quad \min_{B=b_0+\cdots+b_{n-2}x^{n-2}} \left(\max_{-1\leq x\leq 1} |x^n + rx^{n-1} - B(x)| \right)$$

Excerpt from Solotareff's theorems:

$$\forall c_0, c_1 \forall x \exists y\colon\ 0 \leq r \leq 1 \text{ and } (y^3 + ry^2 - c_1 y - c_0)^2$$
$$\geq \left(x^3 + rx^2 - \underbrace{\left(\frac{3}{4} + \frac{r}{2} - \frac{r^2}{4}\right)}_{b_1} x - \underbrace{\left(\frac{r}{4} + \frac{r^2}{6} - \frac{r^3}{108}\right)}_{b_0} \right)^2$$

3. Black Box Linear Algebra

The black box model of a matrix

$A \in \mathbb{K}^{n\times n}$ singular

$\mathbb{K}$ an arbitrary, e.g., finite field

Perform linear algebra operations, e.g., $A^{-1}b$ [Wiedemann 86] with

$$\begin{array}{rl} O(n) & \text{black box calls and} \\ n^2(\log n)^{O(1)} & \text{arithmetic operations in } \mathbb{K} \text{ and} \\ O(n) & \text{intermediate storage for field elements} \end{array}$$

Flurry of recent results

Lambert [1996], Eberly&K [1997]	relationship of Wiedemann and Lanczos approach
Villard [1997]	analysis of *block* Wiedemann algorithm
Giesbrecht [1997]	computation of integral solutions
Giesbrecht&Lobo &Saunders [1997]	certificates for inconsistency

Open Problem 3
Within the resource limitations stated above, compute the characteristic polynomial of a black box matrix. Randomization is allowed (of course!), as is a "Monte Carlo" solution.

Classes of randomized algorithms

Monte Carlo	$\equiv$	always fast, probably correct
Las Vegas	$\equiv$	always correct, probably fast
BPP	$\equiv$	probably correct, probably fast

Why Las Vegas algorithms may be bad for you

```
repeat
        pick random numbers
        compute candidate answer
until check if a solution succeeds
```

A programming bug leads to an infinite loop!

Diophantine solutions
by Giesbrecht:
Find several rational solutions.

$$A(\tfrac{1}{2}x^{[1]}) = b, \quad x^{[1]} \in \mathbb{Z}^n$$
$$A(\tfrac{1}{3}x^{[2]}) = b, \quad x^{[2]} \in \mathbb{Z}^n$$

$$\gcd(2,3) = 1 = 2\cdot 2 - 1\cdot 3$$
$$A(2x^{[1]} - x^{[2]}) = 4b - 3b = b$$

4. Lattice Reduction

$$\pi = \sum_{i=0}^{\infty} \frac{1}{16^i} \left(\frac{4}{8i+1} - \frac{2}{8i+4} - \frac{1}{8i+5} - \frac{1}{8i+6} \right)$$

Derivation by lattice reduction [Bailey&Borwein&Plouffe 1995]

$$\int_0^1 \frac{y^{k-1}}{1-\frac{y^8}{16}} dy = \int_0^1 \sum_{i=0}^{\infty} y^{k-1} \left(\frac{y^8}{16} \right)^i dy = \sum_{i=0}^{\infty} \frac{1}{16^i} \int_0^1 y^{8i+k-1} dy$$

$$= \sum_{i=0}^{\infty} \frac{1}{16^i(8i+k)}$$

Maple takes over

```
> latt := proc(digits)
> local k, j, v, saved_Digits, ltt;
> saved_Digits := Digits; Digits := digits;
> for k from 1 to 8 do
>   v[k] := [];
>   for j from 1 to 10 do v[k] := [op(v[k]), 0]; od;
>   v[k][k] := 1;
>   v[k][10] := trunc(10^digits *
>                     evalf(Int(y^(k-1)/(1-y^8/16),
>                               y=0..1, digits), digits));
> od;
> v[9] := [0,0,0,0,0,0,0,0,1,
>            trunc(evalf(Pi*10^digits,digits+1))];
> ltt := [];
> for k from 1 to 9 do ltt:=[op(ltt),evalm(v[k])];od;
> Digits := saved_Digits;
> RETURN(ltt);
> end:
```

```
> L := latt(25);
```

$$L := [[1, 0, 0, 0, 0, 0, 0, 0, 0, 1007184476414676228644760 0],$$
$$[0, 1, 0, 0, 0, 0, 0, 0, 0, 5064768766674304809559394],$$
$$[0, 0, 1, 0, 0, 0, 0, 0, 0, 3392302452451990725155853],$$
$$[0, 0, 0, 1, 0, 0, 0, 0, 0, 2554128118829953416027570],$$
$$[0, 0, 0, 0, 1, 0, 0, 0, 0, 2050025576364235339441503],$$
$$[0, 0, 0, 0, 0, 1, 0, 0, 0, 1713170706664974589667328],$$
$$[0, 0, 0, 0, 0, 0, 1, 0, 0, 1472019346726350271955981],$$
$$[0, 0, 0, 0, 0, 0, 0, 1, 0, 1290770422751423433458478],$$
$$[0, 0, 0, 0, 0, 0, 0, 0, 1, 3141592653589793238462643 4]]$$

```
> readlib(lattice):
> lattice(L);
```

$$[[-4,0,0,2,1,1,0,0,1,5],[0,-8,-4,-4,0,0,1,0,2,5],$$
$$[-61,582,697,-1253,453,-1003,-347,-396,10,559],$$
$$[-333,966,324,-1656,-56,784,1131,-351,-27,255],$$
$$[429,714,-1591,778,-517,-1215,598,362,-87,398],$$
$$[-1046,-259,-295,-260,1286,393,851,800,252,-1120],$$
$$[494,906,-380,-1389,1120,1845,-1454,-926,-218,400],$$
$$[1001,-1099,422,1766,1405,-376,905,-1277,-394,-30],$$
$$[-1144,491,-637,-736,-1261,-680,-1062,-1257,637,-360]]$$

```
> g := (8*y + 4*y^2 + 4*y^3 - y^6)/(1-y^8/16);
```

$$g := \frac{8\,y+4\,y^2+4\,y^3-y^6}{1-\frac{1}{16}y^8}$$

```
> int(g, y=0..1);
```

$$2\pi$$

Goldreich&Goldwasser&Halevi [1997] public key crypto system

Public key: Lattice basis B (rows B_i are basis vectors).

Private key: *reduced* basis C for lattice spanned by B.

Clear text is represented as a vector x with *small* integer entries.

Encoded message: $y = x + \sum_i r_i B_i$ where $\sum_i r_i B_i$ is a random vector in the lattice.

Decryption based on Babai algorithm [1985] for nearest lattice point: Write $y = \sum_i s_i C_i$ with $s_i \in \mathbb{Q}$. Then $\sum_i \text{nearest-integer}(s_i) C_i$ is a near lattice point, probably $\sum_i r_i B_i$.

Open Problem 4

Devise a public key crypto-system that is based on diophantine linear algebra but that is safe from lattice reduction.

5. Groebner Bases

$$\begin{aligned} f_1 &= x^2 + xy + 2x \qquad\quad + \; y - 1 = 0 \\ f_2 &= x^2 \qquad\quad + 3x - y^2 + 2y - 1 = 0 \\ f_3 &= \qquad\qquad ux \qquad + vy + w \end{aligned}$$

$$(x,y) = (1,-1), (-3,1), (0,1)$$

$$\begin{array}{c|cccccccccc|}
 & x^3 & x^2y & x^2 & xy^2 & xy & x & y^3 & y^2 & y & 1 \\
xf_1 & 1 & 1 & 2 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\
yf_1 & 0 & 1 & 0 & 1 & 2 & 0 & 0 & 1 & -1 & 0 \\
f_1 & 0 & 0 & 1 & 0 & 1 & 2 & 0 & 0 & 1 & -1 \\
xf_2 & 1 & 0 & 3 & -1 & 2 & -1 & 0 & 0 & 0 & 0 \\
yf_2 & 0 & 1 & 0 & 0 & 3 & 0 & -1 & 2 & -1 & 0 \\
f_2 & 0 & 0 & 1 & 0 & 0 & 3 & 0 & -1 & 2 & -1 \\
xyf_3 & 0 & u & 0 & v & w & 0 & 0 & 0 & 0 & 0 \\
xf_3 & 0 & 0 & u & 0 & v & w & 0 & 0 & 0 & 0 \\
yf_3 & 0 & 0 & 0 & 0 & u & 0 & 0 & v & w & 0 \\
f_3 & 0 & 0 & 0 & 0 & 0 & u & 0 & 0 & v & w
\end{array} = \begin{array}{l} (u-v+w) \\ \cdot\,(-3u+v+w) \\ \cdot\,(v+w) \\ \cdot\,(u-v) \end{array}$$

(u-resultant)

Buchberger's algorithm [1967]

S-polynomial construction and reduction correspond to row-reduction in comparable matrices

Faugère's [1997] method: use sparse "symbolic" LU matrix decomposition for performing these row reductions.

Open Problem 5
Compute Gröbner bases approximately by iterative methods for solving systems, such as Gauss&Seidel, conjugate gradient, Newton,...

A solution plugs into numerical software and computes some bases faster than the exact approach; the structure of the bases may be determined, e.g., by modular arithmetic

6. Algorithm Synthesis

Let $\sigma \in \mathbb{K}[\alpha, \beta]/(f, g)$ where $f(\alpha, \beta) = 0$ and $g(\beta) = 0$.
E.g., $\sigma = \sqrt{1+\sqrt{2}} - \sqrt{2} = \alpha - \beta$, $f = \alpha^2 - \beta - 1$, and $g = \beta^2 - 2$.

Task: Compute the minimum polynomial $h(\sigma) = 0:$

$$h(x) = x^m - c_{m-1}x^{m-1} - \cdots - c_0 \in \mathbb{K}[x], \quad m \leq \deg(f) \cdot \deg(g)$$

The coefficient vectors $\overrightarrow{\sigma^i}$ of σ^i mod $(f(\alpha, \beta), g(\beta))$ satisfy

$$\forall\, j \geq 0\colon\ \overrightarrow{\sigma^{m+j}} = c_{m-1}\overrightarrow{\sigma^{m-1+j}} + \cdots + c_0\overrightarrow{\sigma^j}$$

Any non-trivial linear projection $\mathcal{L}(\overrightarrow{\sigma^i})$ preserves the linear recursion because h is irreducible.

Power Projections = Transposed Modular Polyn Composition

Linear projections of powers

$$\left[\mathcal{L}(\overrightarrow{\sigma^0})\ \mathcal{L}(\overrightarrow{\sigma^1})\mathcal{L}(\overrightarrow{\sigma^2})\ \ldots\right] = \left[u_0\ u_1\ \ldots\ u_{n-1}\right] \cdot \underbrace{\left[\overrightarrow{\sigma^0} \mid \overrightarrow{\sigma^1} \mid \overrightarrow{\sigma^2} \mid \ldots\right]}_{A}$$

Modular polynomial composition

$$w(z) = w_0 + w_1 z + w_2 z^2 + \cdots \longmapsto w(\sigma) \bmod (f(\alpha,\beta), g(\beta))$$

$$\overrightarrow{w(\sigma)} = \underbrace{\left[\overrightarrow{\sigma^0} \mid \overrightarrow{\sigma^1} \mid \overrightarrow{\sigma^2} \mid \ldots\right]}_{A} \cdot \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \end{bmatrix}$$

By Tellegen's Theorem [1960] the problems can be solved equally fast

Transposed Modular Polynomial Multiplication in NTL

1. $T_1 \leftarrow \mathrm{FFT}^{-1}(\mathrm{RED}_k(g))$
2. $T_2 \leftarrow T_1 \cdot S_2$
3. $v \leftarrow -\mathrm{CRT}_{0\ldots n-2}(\mathrm{FFT}(T_2))$
4. $T_2 \leftarrow \mathrm{FFT}^{-1}(\mathrm{RED}_{k+1}(x^{n-1} \cdot v))$
5. $T_2 \leftarrow T_2 \cdot S_3$
6. $T_1 \leftarrow T_1 \cdot S_4$
7. Replace T_1 by the 2^{k+1}-point residue table whose j-th column $(0 \leq j < 2^{k+1})$ is 0 if j is odd, and is column number $j/2$ of T_1 if j is even.
8. $T_2 \leftarrow T_2 + T_1$
9. $u \leftarrow \mathrm{CRT}_{0\ldots n-1}(\mathrm{FFT}(T_2))$

"we offer no other proof of correctness other than the validity of this transformation technique (and the fact that it does indeed work in practice)" [Shoup 1994]

Open Problem 6
With inputs $A \in \mathbb{K}^{m\times n}$ *and* $y \in \mathbb{K}^n$ *you are given an algorithm for* $A \cdot y$ *that uses* $T(m,n)$ *arithmetic field operations and* $S(m,n)$ *auxiliary space.*

Show how to construct an algorithm for $A^T \cdot z$ *where* $z \in \mathbb{K}^m$ *that uses* $O(T(m,n))$ *time and* $O(S(m,n))$ *space.*

Your construction must be applicable to practical problems.

7. Knuth's Critique of Asymptotically Fast Methods

End of §4.6.2 Factorization of Polynomials (page 455)

> *"The asymptotically best algorithms frequently turn out to be worst on all problems for which they are used.*
>
> — D. G. CANTOR and H. ZASSENHAUS (1981)"

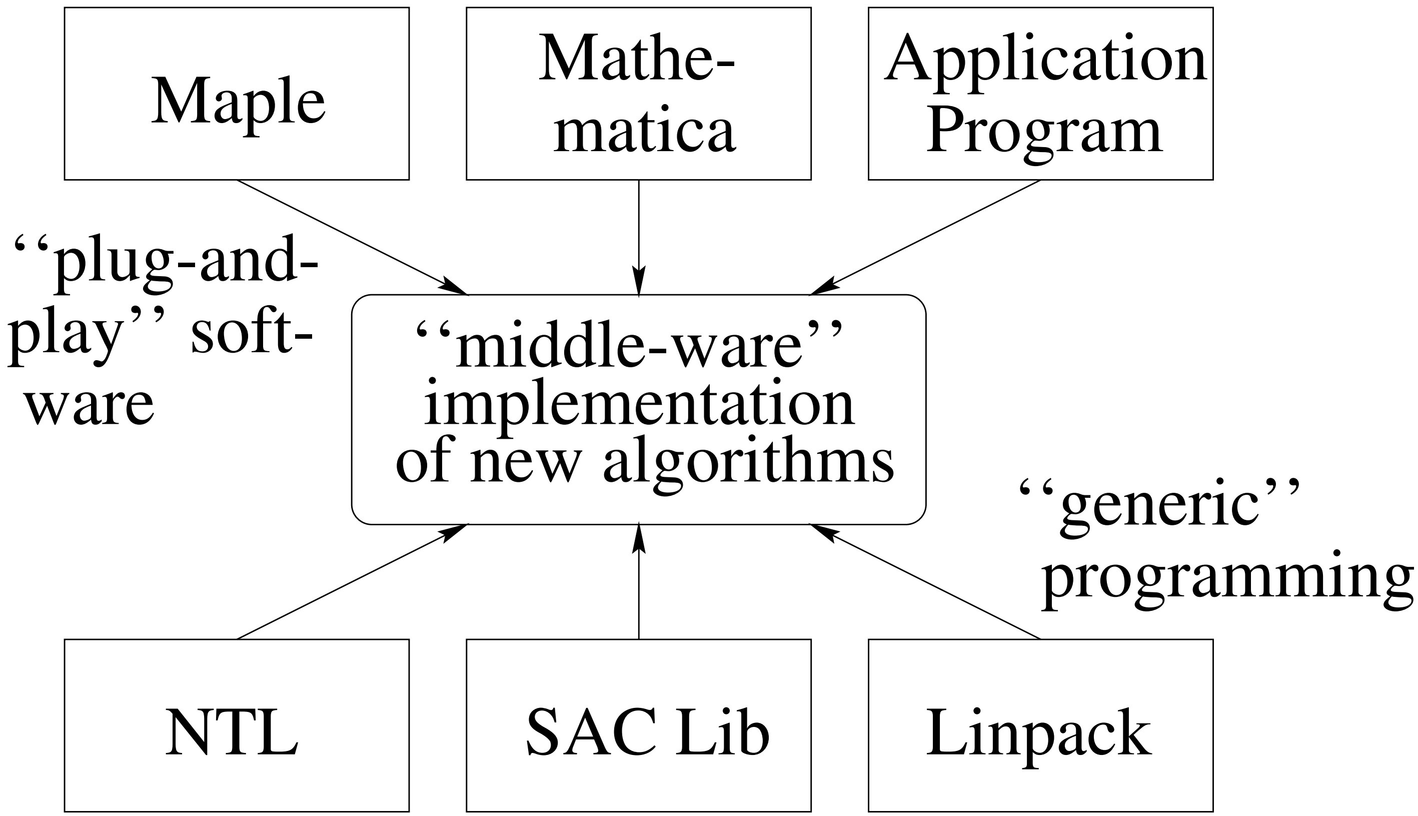
Answer to Exercise 70 of §4.6.4 (page 718)

"E. Kaltofen has in fact constructed a determinant evaluation algorithm that requires only $O(n^{(\omega+4)/2+\varepsilon})$ additions, subtractions, and multiplications [ISSAC 92]. Of course such asymptotically **'**fast**'** matrix multiplication is strictly of theoretical interest."

Open Problem 7

Convince Donald Knuth that these asymptotically fast methods are of practical value. If he pays you \$2.56 for this technical error, you have solved this problem.

8. Plug-And-Play Components



Problem solving environ’s: end-user can easily custom-make symbolic software

Example: FoxBox [Díaz and K 1998]

```
# Call FoxBox server from Maple
> SymToeQ  := BlackBoxSymToe( BBNET_Q, 4, -1, 1.0 ):
> SymToeZP := BlackBoxSymToe( BBNET_ZP, 4, -1, 1.0 ):
> FactorsQ := BlackBoxFactors( BBNET_Q, SymToeQ, Mod, 1.0,
                               Seed ):
> FactorsZP := BlackBoxHomomorphicMap( BBNET_FACS, FactorsQ,
                                       SymToeZP ):
```

```
// construct factors of a symmetric Toeplitz determinant in C++
typedef BlackBoxSymToeDet< SaclibQ, SaclibQX > BBSymToeDetQ;
typedef BlackBoxFactors< SaclibQ, SaclibQX,
                                BBSymToeDetQ > BBFactorsQ;

BBSymToeDetQ SymToeDetQ( N );
BBFactorsQ   FactorsQ( SymToeDetQ, Probab, Seed, &MPCard );
```

Software Design Issues

Plug-and-play

- Standard representation for transfer: MP, OpenMath, MathML
- Byte code for constructing objects vs. parse trees
- Visual programming environments for composition

Generic Programming

- Common object interface (wrapper classes),
  e.g., `K::random_generator(500)`
- Storage management vs. garbage collection
- Algorithmic shortcuts into the basic modules

Open Problem 8

Devise a plug-and-play and generic programming methodology for symbolic mathematical computation that is widely adopted by the experts in algorithm design, the commercial symbolic software producers, and the outsider users.

“Designing a system that plugs in someone else’s is difficult”

[K 1997]

“Designing a system that someone else can plug-in is difficult”

[Hong 1997]

9. Another "Killer" Application (KA)

KA for the Macintosh: Document preparation

KA for the PC: Spreadsheets

KA application for supercomputers: Weather forecasting

KA for mainframes: Social security system

KA for symbolic software: Calculus teaching

Open Problem 9
Besides math education, find another so-called "killer" application for symbolic computation.

The problem is solved when the new application makes the software written for it a commercial success.

Summary

1. Nearby multivariate polynomials that factor over $\mathbb{C}$
2. Solotareff's problem on a computer
3. Characteristic polynomial of a black box matrix
4. Lattice reduction-safe GGH crypto-system
5. Gröbner bases via iterative methods
6. Space&time efficient transposition principle
7. Knuth's opinion on asymptotically fast algorithms
8. Plug-and-play and generic programming methodology for symbolic computation
9. Another "killer" application besides education